LAB 06

MiniMax Algorithm

Lab Task

1. Implement Minimax in Tic Tac Toe game

Code

# Initialize empty board

board = [" " for \_ in range(9)]

# Print the board in a simple format

def print\_board():

    for i in range(0, 9, 3):

        print("| " + " | ".join(board[i:i+3]) + " |")

# Check if there is a winner or the game is a draw

def check\_winner(player):

    win\_patterns = [

        [0, 1, 2], [3, 4, 5], [6, 7, 8],  # Rows

        [0, 3, 6], [1, 4, 7], [2, 5, 8],  # Columns

        [0, 4, 8], [2, 4, 6]              # Diagonals

    ]

    for pattern in win\_patterns:

        if all(board[i] == player for i in pattern):

            return True

    return False

# MiniMax function to find the best move

def minimax(is\_ai):

    if check\_winner("O"):

        return 1

    if check\_winner("X"):

        return -1

    if " " not in board:

        return 0

    best\_score = float('-inf') if is\_ai else float('inf')

    for i in range(9):

        if board[i] == " ":

            board[i] = "O" if is\_ai else "X"

            score = minimax(not is\_ai)

            board[i] = " "

            best\_score = max(score, best\_score) if is\_ai else min(score, best\_score)

    return best\_score

# Find the best move for AI

def best\_move():

    best\_score, move = float('-inf'), -1

    for i in range(9):

        if board[i] == " ":

            board[i] = "O"

            score = minimax(False)

            board[i] = " "

            if score > best\_score:

                best\_score, move = score, i

    board[move] = "O"

# Main game loop

def play\_game():

    print\_board()

    while " " in board:

        player\_move = int(input("Enter your move (0-8): "))

        if board[player\_move] == " ":

            board[player\_move] = "X"

            if check\_winner("X"):

                print\_board()

                print("You win!")

                return

            best\_move()

            print\_board()

            if check\_winner("O"):

                print("AI wins!")

                return

        else:

            print("Cell taken, try again.")

    print("It's a draw!")

play\_game()

Output



